

Bandwidth-Delay Product

Roman Dunaytsev

Department of Communications Networks and Data Transmission
Saint-Petersburg State University of Telecommunications

roman.dunaytsev@spbgut.ru

Lab № 2

Outline

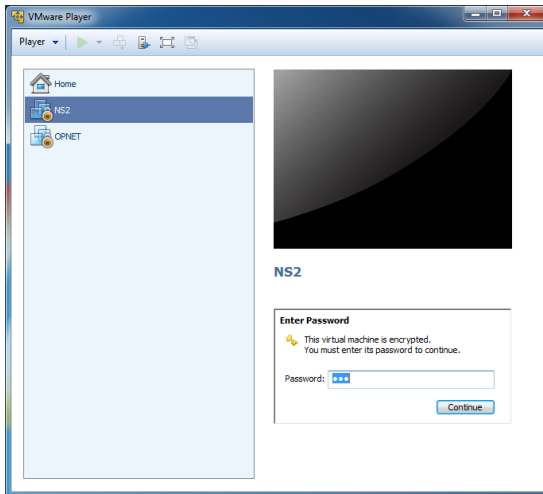
- 1 VMware Player
- 2 Sizing router buffers
- 3 Bandwidth-delay product

Outline

- 1 VMware Player
- 2 Sizing router buffers
- 3 Bandwidth-delay product

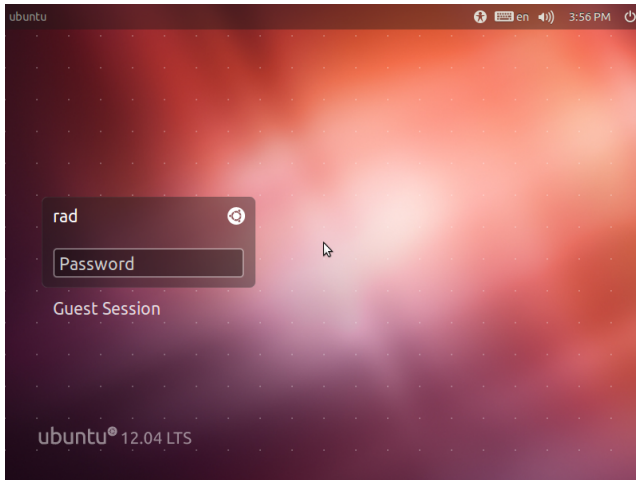
VMware Player

- Virtual machine: **NS2**
- Password to run: **123**



VMware Player (con'd)

- Password: 123
- Or run **Guest Session**



Outline

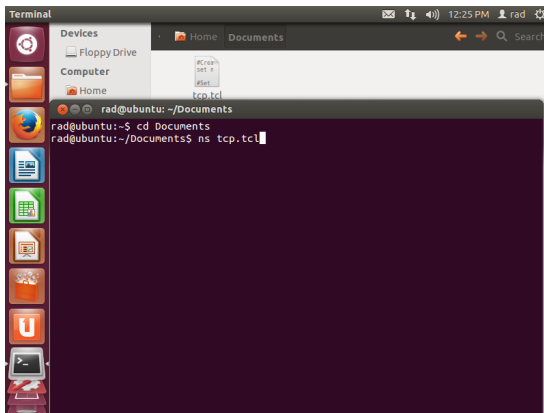
1 VMware Player

2 Sizing router buffers

3 Bandwidth-delay product

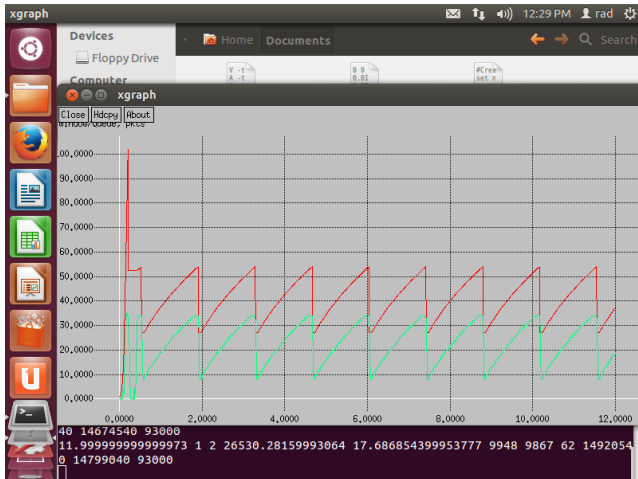
Sizing Router Buffers

- Open **Terminal**: <Ctrl>+<Alt>+<T>
- Change directory: **cd Documents**
- Run simulation: **ns tcp.tcl**



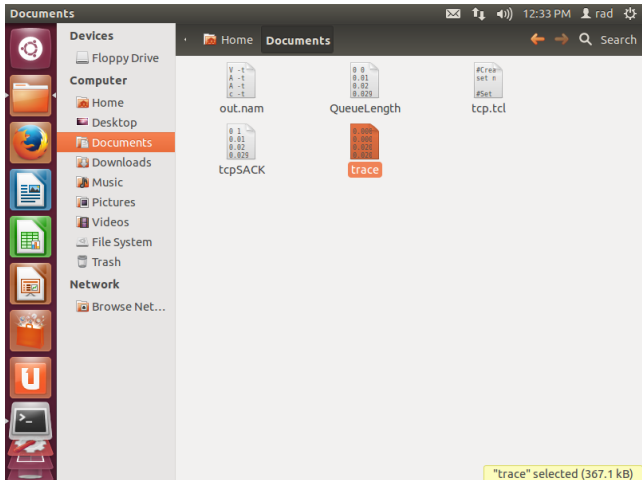
Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = **36 packets**



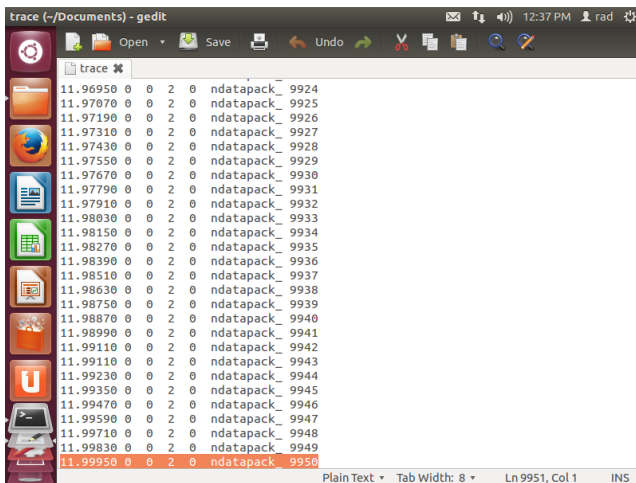
Sizing Router Buffers (cont'd)

- Open: **trace**



Sizing Router Buffers (cont'd)

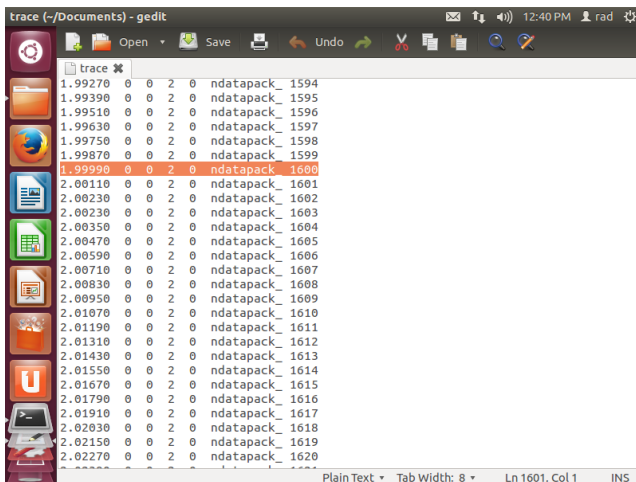
- Time: from 2 to 12



```
trace (~/.Documents) - gedit
11.96950 0 0 2 0 ndatapack_ 9924
11.97070 0 0 2 0 ndatapack_ 9925
11.97190 0 0 2 0 ndatapack_ 9926
11.97310 0 0 2 0 ndatapack_ 9927
11.97430 0 0 2 0 ndatapack_ 9928
11.97550 0 0 2 0 ndatapack_ 9929
11.97670 0 0 2 0 ndatapack_ 9930
11.97790 0 0 2 0 ndatapack_ 9931
11.97910 0 0 2 0 ndatapack_ 9932
11.98030 0 0 2 0 ndatapack_ 9933
11.98150 0 0 2 0 ndatapack_ 9934
11.98270 0 0 2 0 ndatapack_ 9935
11.98390 0 0 2 0 ndatapack_ 9936
11.98510 0 0 2 0 ndatapack_ 9937
11.98630 0 0 2 0 ndatapack_ 9938
11.98750 0 0 2 0 ndatapack_ 9939
11.98870 0 0 2 0 ndatapack_ 9940
11.98990 0 0 2 0 ndatapack_ 9941
11.99110 0 0 2 0 ndatapack_ 9942
11.99110 0 0 2 0 ndatapack_ 9943
11.99230 0 0 2 0 ndatapack_ 9944
11.99350 0 0 2 0 ndatapack_ 9945
11.99470 0 0 2 0 ndatapack_ 9946
11.99590 0 0 2 0 ndatapack_ 9947
11.99710 0 0 2 0 ndatapack_ 9948
11.99830 0 0 2 0 ndatapack_ 9949
11.99950 0 0 2 0 ndatapack_ 9950
Plain Text Tab Width: 8 Ln 9951, Col 1 INS
```

Sizing Router Buffers (cont'd)

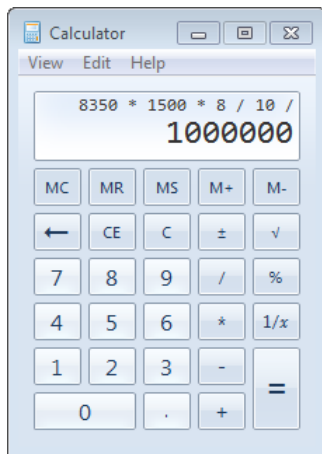
- Time: from 2 to 12



```
trace (~/.Documents) - gedit
1.99270 0 0 2 0 ndatapack_ 1594
1.99390 0 0 2 0 ndatapack_ 1595
1.99510 0 0 2 0 ndatapack_ 1596
1.99630 0 0 2 0 ndatapack_ 1597
1.99750 0 0 2 0 ndatapack_ 1598
1.99870 0 0 2 0 ndatapack_ 1599
1.99990 0 0 2 0 ndatapack_ 1600
2.00110 0 0 2 0 ndatapack_ 1601
2.00230 0 0 2 0 ndatapack_ 1602
2.00230 0 0 2 0 ndatapack_ 1603
2.00350 0 0 2 0 ndatapack_ 1604
2.00470 0 0 2 0 ndatapack_ 1605
2.00590 0 0 2 0 ndatapack_ 1606
2.00710 0 0 2 0 ndatapack_ 1607
2.00830 0 0 2 0 ndatapack_ 1608
2.00950 0 0 2 0 ndatapack_ 1609
2.01070 0 0 2 0 ndatapack_ 1610
2.01190 0 0 2 0 ndatapack_ 1611
2.01310 0 0 2 0 ndatapack_ 1612
2.01430 0 0 2 0 ndatapack_ 1613
2.01550 0 0 2 0 ndatapack_ 1614
2.01670 0 0 2 0 ndatapack_ 1615
2.01790 0 0 2 0 ndatapack_ 1616
2.01910 0 0 2 0 ndatapack_ 1617
2.02030 0 0 2 0 ndatapack_ 1618
2.02150 0 0 2 0 ndatapack_ 1619
2.02270 0 0 2 0 ndatapack_ 1620
```

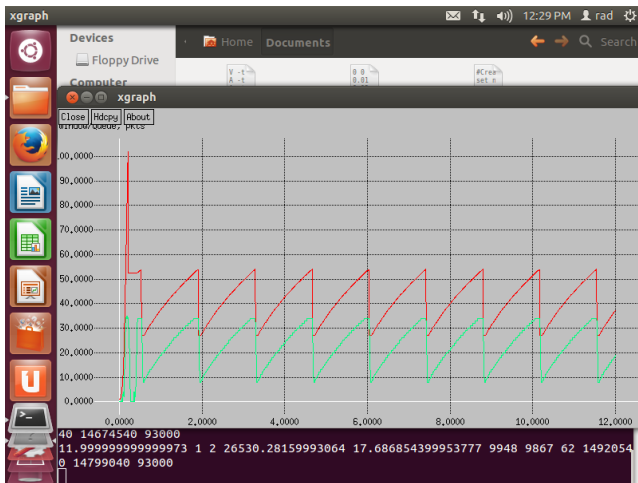
Sizing Router Buffers (cont'd)

- Packets sent: $9950 - 1600 = 8350$
- Throughput: 10.02 Mb/s



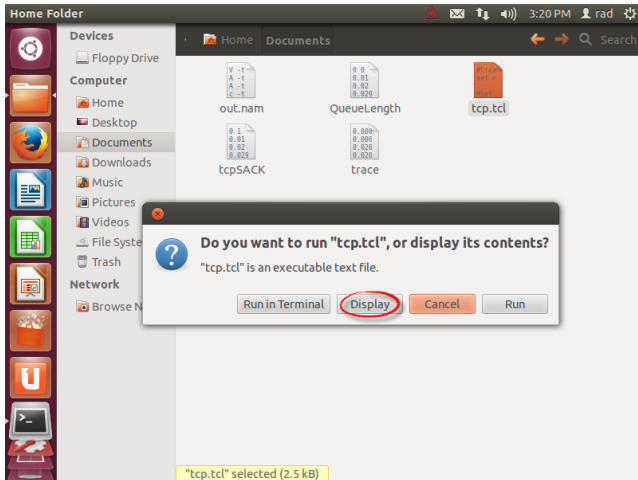
Sizing Router Buffers (cont'd)

- Packets lost: 7
- Packet loss rate: $7 / 8350 = 0.00084$



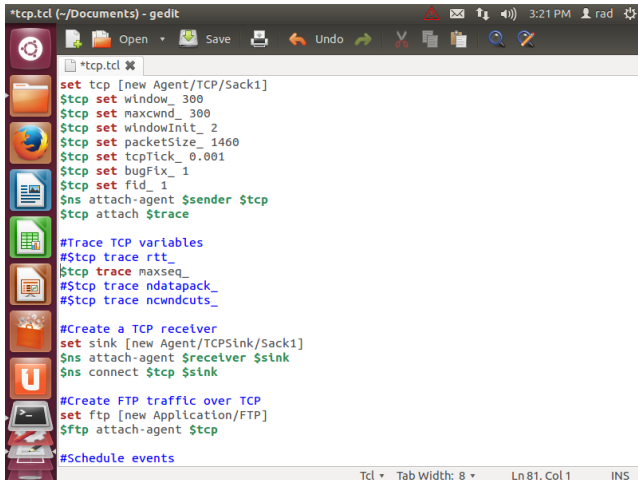
Sizing Router Buffers (cont'd)

- Edit: **tcp.tcl**



Sizing Router Buffers (cont'd)

- Trace: `maxseq_`



The screenshot shows a gedit editor window titled '*tcp.tcl (~/.Documents) - gedit'. The window contains a Tcl script for simulating a TCP connection. The script includes comments in blue and code in green. The code sets up a TCP agent, configures its parameters (window, maxwnd, windowInit, packetSize, tcpTick, bugFix, fid), attaches it to a sender, and traces various variables. It also creates a TCP receiver (sink) and connects the sender to it. Finally, it creates an FTP application over the TCP connection and schedules events.

```
*tcp.tcl (~/.Documents) - gedit
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
$tcp attach $trace

#Trace TCP variables
#$tcp trace rtt_
#$tcp trace maxseq_
#$tcp trace ndatapack_
#$tcp trace ncwndcuts_

#Create a TCP receiver
set sink [new Agent/TCPSink/Sack1]
$ns attach-agent $receiver $sink
$ns connect $tcp $sink

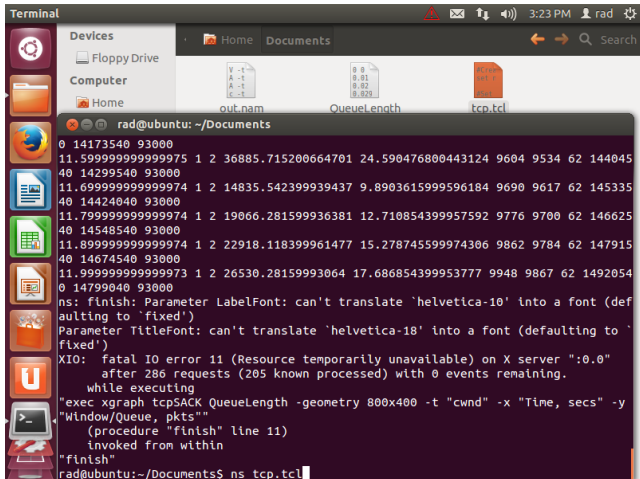
#Create FTP traffic over TCP
set ftp [new Application/FTP]
$ftp attach-agent $tcp

#Schedule events
```

The status bar at the bottom of the window shows 'Tcl', 'Tab Width: 8', 'Ln 81, Col 1', and 'INS'.

Sizing Router Buffers (cont'd)

- Rerun: **tcp.tcl**

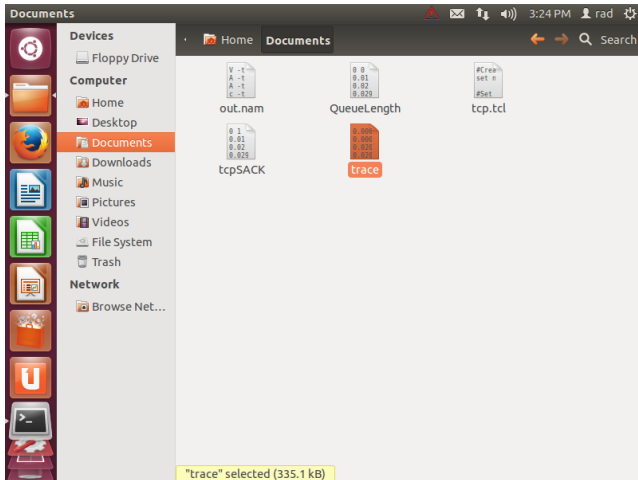


The image shows a terminal window titled "Terminal" with a sidebar on the left containing icons for various applications. The main window displays the command prompt "rad@ubuntu: ~/Documents" and the execution of the script "tcp.tcl". The output of the script is a series of lines representing network traffic statistics, including source and destination IP addresses, sequence numbers, and packet counts. Following the script execution, several error messages are displayed, including a warning about font translation for the terminal window and a fatal XIO error related to the X server.

```
rad@ubuntu: ~/Documents
0 14173540 93000
11.599999999999997 1 2 36885.715200664701 24.590476800443124 9604 9534 62 144045
40 14299540 93000
11.699999999999997 1 2 14835.542399939437 9.8903615999596184 9690 9617 62 145335
40 14424040 93000
11.799999999999997 1 2 19066.281599936381 12.710854399957592 9776 9700 62 146625
40 14548540 93000
11.899999999999997 1 2 22918.118399961477 15.278745599974306 9862 9784 62 147915
40 14674540 93000
11.999999999999997 1 2 26530.28159993064 17.686854399953777 9948 9867 62 1492054
0 14799040 93000
ns: finish: Parameter LabelFont: can't translate 'helvetica-10' into a font (def
aulting to 'fixed')
Parameter TitleFont: can't translate 'helvetica-18' into a font (defaulting to '
fixed')
XIO: fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
after 286 requests (205 known processed) with 0 events remaining.
while executing
"exec xgraph tcpSACK QueueLength -geometry 800x400 -t "cwnd" -x "Time, secs" -y
"Window/Queue, pkts"
(procedure "finish" line 11)
invoked from within
"finish"
rad@ubuntu:~/Documents$ ns tcp.tcl
```

Sizing Router Buffers (cont'd)

- Open: **trace**



Sizing Router Buffers (cont'd)

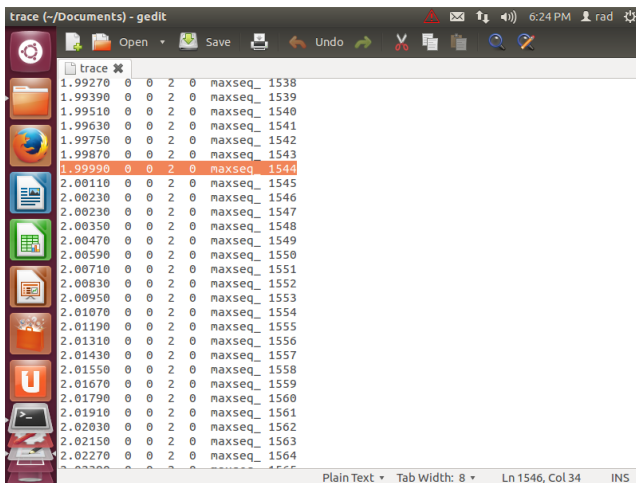
- Time: from 2 to 12

```
trace (~/.Documents) - gedit
11.96950 0 0 2 0 maxseq_ 9861
11.97070 0 0 2 0 maxseq_ 9862
11.97190 0 0 2 0 maxseq_ 9863
11.97310 0 0 2 0 maxseq_ 9864
11.97430 0 0 2 0 maxseq_ 9865
11.97550 0 0 2 0 maxseq_ 9866
11.97670 0 0 2 0 maxseq_ 9867
11.97790 0 0 2 0 maxseq_ 9868
11.97910 0 0 2 0 maxseq_ 9869
11.98030 0 0 2 0 maxseq_ 9870
11.98150 0 0 2 0 maxseq_ 9871
11.98270 0 0 2 0 maxseq_ 9872
11.98390 0 0 2 0 maxseq_ 9873
11.98510 0 0 2 0 maxseq_ 9874
11.98630 0 0 2 0 maxseq_ 9875
11.98750 0 0 2 0 maxseq_ 9876
11.98870 0 0 2 0 maxseq_ 9877
11.98990 0 0 2 0 maxseq_ 9878
11.99110 0 0 2 0 maxseq_ 9879
11.99110 0 0 2 0 maxseq_ 9880
11.99230 0 0 2 0 maxseq_ 9881
11.99350 0 0 2 0 maxseq_ 9882
11.99470 0 0 2 0 maxseq_ 9883
11.99590 0 0 2 0 maxseq_ 9884
11.99710 0 0 2 0 maxseq_ 9885
11.99830 0 0 2 0 maxseq_ 9886
11.99950 0 0 2 0 maxseq_ 9887
```

Plain Text ▾ Tab Width: 8 ▾ Ln 9889, Col 1 INS

Sizing Router Buffers (cont'd)

- Time: from 2 to 12

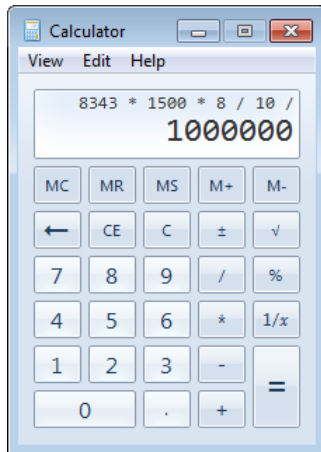


```
trace (~/.Documents) - gedit
1.99270 0 0 2 0 maxseq_ 1538
1.99390 0 0 2 0 maxseq_ 1539
1.99510 0 0 2 0 maxseq_ 1540
1.99630 0 0 2 0 maxseq_ 1541
1.99750 0 0 2 0 maxseq_ 1542
1.99870 0 0 2 0 maxseq_ 1543
1.99990 0 0 2 0 maxseq_ 1544
2.00110 0 0 2 0 maxseq_ 1545
2.00230 0 0 2 0 maxseq_ 1546
2.00230 0 0 2 0 maxseq_ 1547
2.00350 0 0 2 0 maxseq_ 1548
2.00470 0 0 2 0 maxseq_ 1549
2.00590 0 0 2 0 maxseq_ 1550
2.00710 0 0 2 0 maxseq_ 1551
2.00830 0 0 2 0 maxseq_ 1552
2.00950 0 0 2 0 maxseq_ 1553
2.01070 0 0 2 0 maxseq_ 1554
2.01190 0 0 2 0 maxseq_ 1555
2.01310 0 0 2 0 maxseq_ 1556
2.01430 0 0 2 0 maxseq_ 1557
2.01550 0 0 2 0 maxseq_ 1558
2.01670 0 0 2 0 maxseq_ 1559
2.01790 0 0 2 0 maxseq_ 1560
2.01910 0 0 2 0 maxseq_ 1561
2.02030 0 0 2 0 maxseq_ 1562
2.02150 0 0 2 0 maxseq_ 1563
2.02270 0 0 2 0 maxseq_ 1564
2.02390 0 0 2 0 maxseq_ 1565
```

Plain Text ▾ Tab Width: 8 ▾ Ln 1546, Col 34 INS

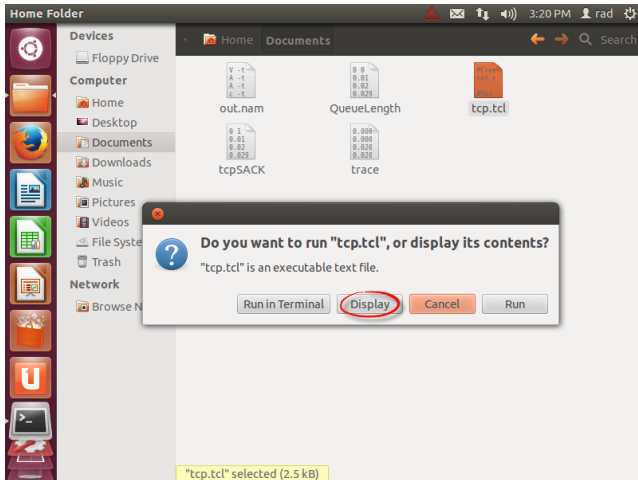
Sizing Router Buffers (cont'd)

- Packets received: $9887 - 1544 = 8343$
- Goodput: **10.02 Mb/s**



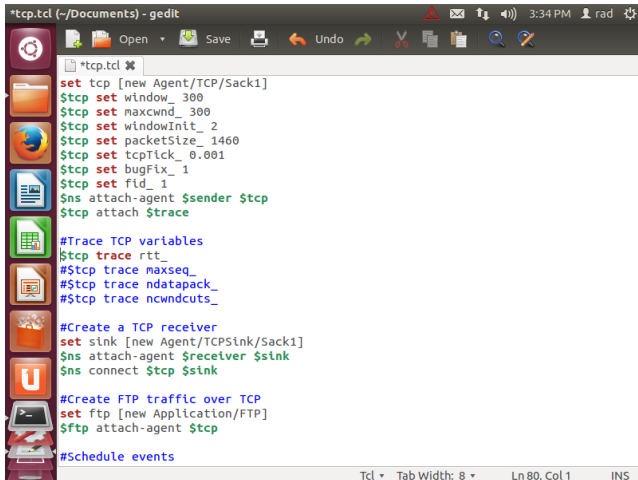
Sizing Router Buffers (cont'd)

- Edit: **tcp.tcl**



Sizing Router Buffers (cont'd)

- Trace: `rtt_`



The screenshot shows a NetSim TCL script editor window titled '*tcp.tcl (~/Documents) - gedit'. The script configures a TCP agent, sets various parameters, attaches a trace, creates a receiver, and schedules events.

```
*tcp.tcl (~/Documents) - gedit
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fld_ 1
$ns attach-agent $sender $tcp
$tcp attach $trace

#Trace TCP variables
$tcp trace rtt_
#$tcp trace maxseq_
#$tcp trace ndatapack_
#$tcp trace ncwndcuts_

#Create a TCP receiver
set sink [new Agent/TCPSink/Sack1]
$ns attach-agent $receiver $sink
$ns connect $tcp $sink

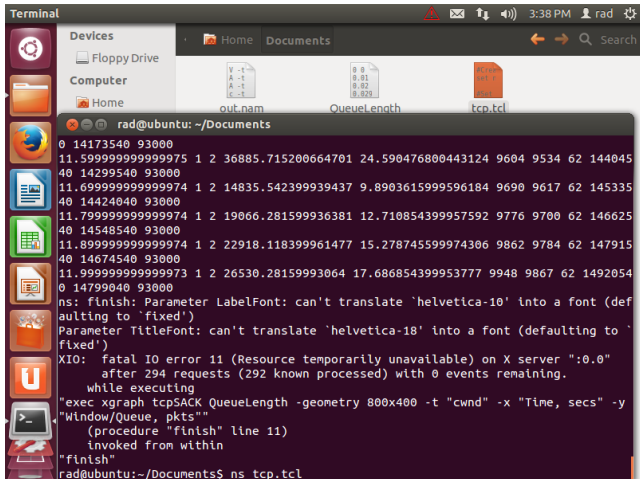
#Create FTP traffic over TCP
set ftp [new Application/FTP]
$ftp attach-agent $tcp

#Schedule events
```

The status bar at the bottom indicates 'Tcl', 'Tab Width: 8', 'Ln 80, Col 1', and 'INS'.

Sizing Router Buffers (cont'd)

- Rerun: **tcp.tcl**

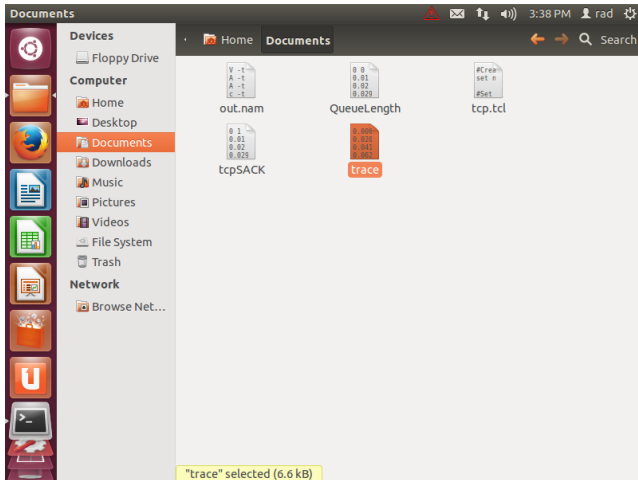


The image shows a terminal window titled "Terminal" with a sidebar on the left containing icons for various applications. The main window displays the output of a script named "tcp.tcl". The output consists of several lines of numerical data, followed by error messages from the X server and the "ns" (Network Simulator) application. The data lines appear to be performance metrics, possibly related to buffer sizing. The error messages indicate that the X server cannot translate font names like "helvetica-10" and "helvetica-18" into a font, and that the "ns" application is experiencing a fatal IO error.

```
rad@ubuntu: ~/Documents
0 14173540 93000
11.599999999999975 1 2 36885.715200664701 24.590476800443124 9604 9534 62 144045
40 14299540 93000
11.699999999999974 1 2 14835.542399939437 9.8903615999596184 9690 9617 62 145335
40 14424040 93000
11.799999999999974 1 2 19066.281599936381 12.710854399957592 9776 9700 62 146625
40 14548540 93000
11.899999999999974 1 2 22918.118399961477 15.278745599974306 9862 9784 62 147915
40 14674540 93000
11.999999999999973 1 2 26530.28159993064 17.686854399953777 9948 9867 62 1492054
0 14799040 93000
ns: finish: Parameter LabelFont: can't translate 'helvetica-10' into a font (def
aulting to 'fixed')
Parameter TitleFont: can't translate 'helvetica-18' into a font (defaulting to '
fixed')
XIO: fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
after 294 requests (292 known processed) with 0 events remaining.
while executing
"exec xgraph tcpSACK QueueLength -geometry 800x400 -t "cwnd" -x "Time, secs" -y
"Window/Queue, pkts"
(procedure "finish" line 11)
invoked from within
"finish"
rad@ubuntu:~/Documents$ ns tcp.tcl
```

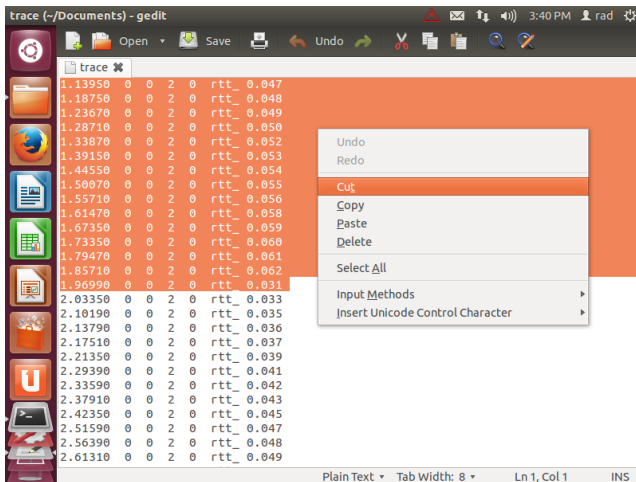
Sizing Router Buffers (cont'd)

- Open: **trace**



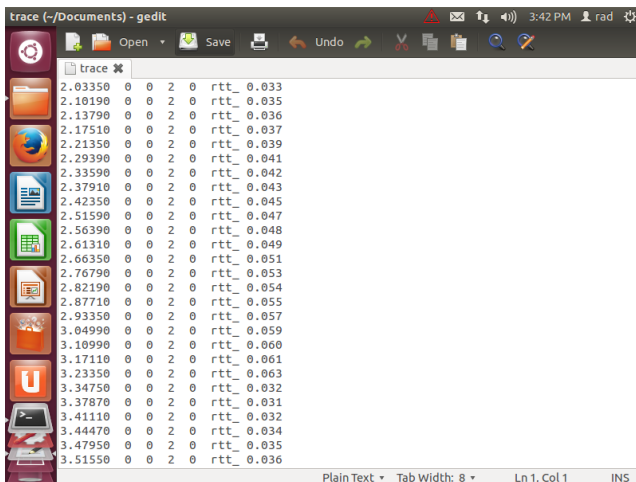
Sizing Router Buffers (cont'd)

- Cut: first 2 seconds



Sizing Router Buffers (cont'd)

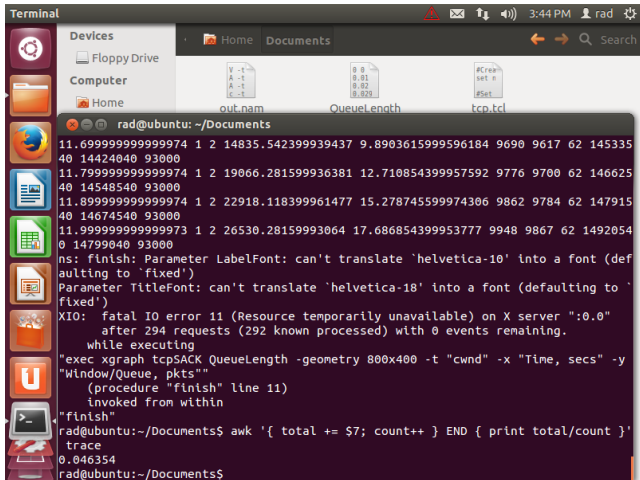
- Save: **trace**



```
trace (~/.Documents) - gedit
Open Save Undo
trace x
2.03350 0 0 2 0 rtt_ 0.033
2.10190 0 0 2 0 rtt_ 0.035
2.13790 0 0 2 0 rtt_ 0.036
2.17510 0 0 2 0 rtt_ 0.037
2.21350 0 0 2 0 rtt_ 0.039
2.29390 0 0 2 0 rtt_ 0.041
2.33590 0 0 2 0 rtt_ 0.042
2.37910 0 0 2 0 rtt_ 0.043
2.42350 0 0 2 0 rtt_ 0.045
2.51590 0 0 2 0 rtt_ 0.047
2.56390 0 0 2 0 rtt_ 0.048
2.61310 0 0 2 0 rtt_ 0.049
2.66350 0 0 2 0 rtt_ 0.051
2.76790 0 0 2 0 rtt_ 0.053
2.82190 0 0 2 0 rtt_ 0.054
2.87710 0 0 2 0 rtt_ 0.055
2.93350 0 0 2 0 rtt_ 0.057
3.04990 0 0 2 0 rtt_ 0.059
3.10990 0 0 2 0 rtt_ 0.060
3.17110 0 0 2 0 rtt_ 0.061
3.23350 0 0 2 0 rtt_ 0.063
3.34750 0 0 2 0 rtt_ 0.032
3.37870 0 0 2 0 rtt_ 0.031
3.41110 0 0 2 0 rtt_ 0.032
3.44470 0 0 2 0 rtt_ 0.034
3.47950 0 0 2 0 rtt_ 0.035
3.51550 0 0 2 0 rtt_ 0.036
Plain Text Tab Width: 8 Ln 1, Col 1 INS
```

Sizing Router Buffers (cont'd)

- RTT average: **0.046 s**

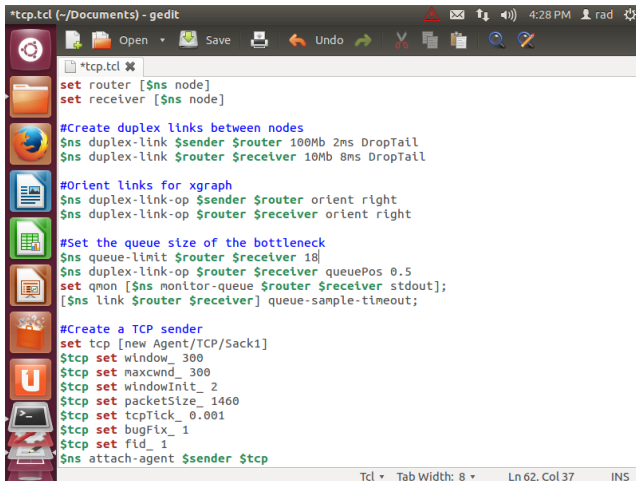


The image shows a terminal window on a Linux system (Ubuntu) with a desktop environment. The terminal output displays the results of a network simulation. It includes several lines of statistics, such as 'ns: finish: Parameter LabelFont: can't translate 'helvetica-10' into a font (defaulting to 'fixed')' and 'Parameter TitleFont: can't translate 'helvetica-18' into a font (defaulting to 'fixed')'. It also shows an error message: 'XIO: fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0" after 294 requests (292 known processed) with 0 events remaining. while executing'. The terminal then shows the execution of 'xgraph tcpSACK QueueLength -geometry 800x400 -t "cwnd" -x "Time, secs" -y "Window/Queue, pkts"' and the output of 'awk' and 'trace' commands. The final output is '0.046354', which corresponds to the RTT average mentioned in the text above.

```
rad@ubuntu: ~/Documents
11.6999999999999974 1 2 14835.542399939437 9.8903615999596184 9690 9617 62 145335
40 14424040 93000
11.7999999999999974 1 2 19066.281599936381 12.710854399957592 9776 9700 62 146625
40 14548540 93000
11.8999999999999974 1 2 22918.118399961477 15.278745599974306 9862 9784 62 147915
40 14674540 93000
11.9999999999999973 1 2 26530.28159993064 17.686854399953777 9948 9867 62 1492054
0 14799040 93000
ns: finish: Parameter LabelFont: can't translate 'helvetica-10' into a font (def
aulting to 'fixed')
Parameter TitleFont: can't translate 'helvetica-18' into a font (defaulting to '
fixed')
XIO: fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
after 294 requests (292 known processed) with 0 events remaining.
while executing
"exec xgraph tcpSACK QueueLength -geometry 800x400 -t "cwnd" -x "Time, secs" -y
"Window/Queue, pkts"
(procedure "finish" line 11)
invoked from within
"finish"
rad@ubuntu:~/Documents$ awk '{ total += $7; count++ } END { print total/count }'
trace
0.046354
rad@ubuntu:~/Documents$
```

Sizing Router Buffers (cont'd)

- Edit: **tcp.tcl**
- Bottleneck router buffer size = **18 packets**



```
*tcp.tcl
set router [$ns node]
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

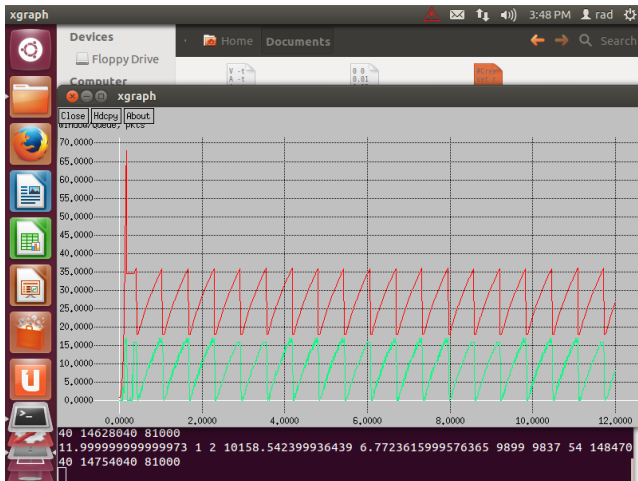
#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 18
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
```

Sizing Router Buffers (cont'd)

- Time: **from 2 to 12**
- Throughput = ? Packet loss rate = ? Goodput = ? Mean RTT = ?

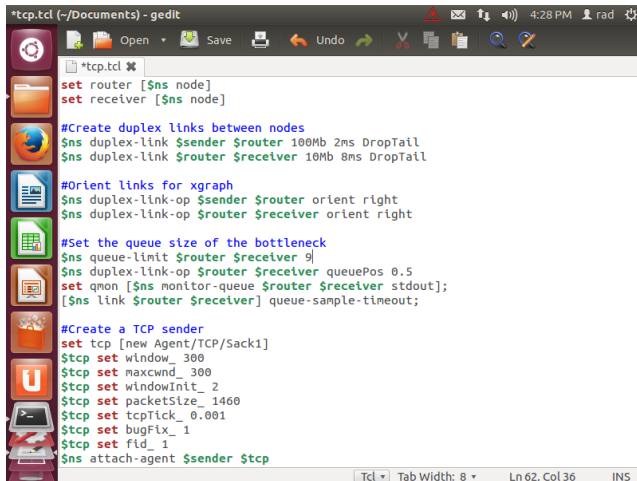


Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = **36 packets**
 - Throughput = 10.02 Mb/s
 - Goodput = 10.02 Mb/s
 - Packet loss rate = 0.00084
 - RTT = 0.046 s
- Bottleneck router buffer size = **18 packets**
 - Throughput = 9.99 Mb/s
 - Goodput = 9.99 Mb/s
 - Packet loss rate = $16 / (9901 - 1579) = 0.00192$
 - RTT = 0.030 s

Sizing Router Buffers (cont'd)

- Edit: **tcp.tcl**
- Bottleneck router buffer size = **9 packets**



```
*tcp.tcl ✖
set router [$ns node]
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

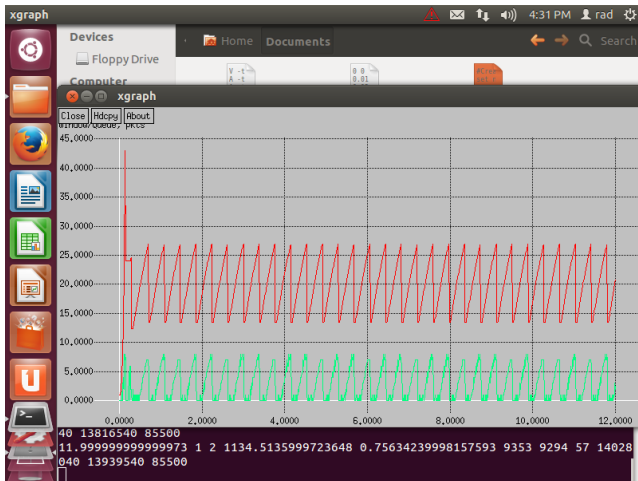
#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 9
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
```

Tcl ▾ Tab Width: 8 ▾ Ln 62, Col 36 INS

Sizing Router Buffers (cont'd)

- Time: **from 2 to 12**
- Throughput = ? Packet loss rate = ? Goodput = ? Mean RTT = ?

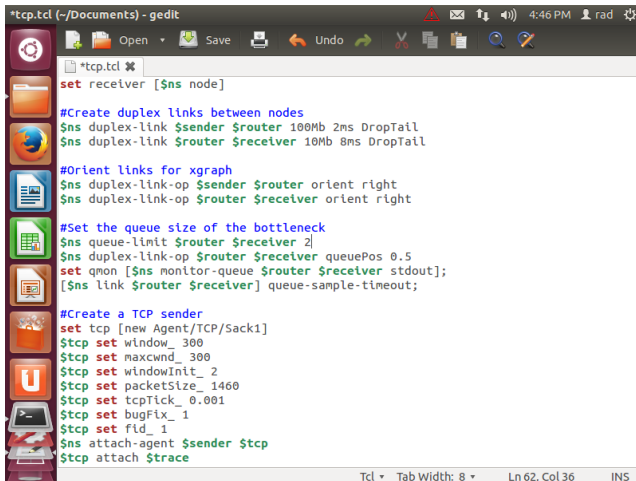


Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = **36 packets**
 - Throughput = 10.02 Mb/s
 - Goodput = 10.02 Mb/s
 - Packet loss rate = 0.00084
 - RTT = 0.046 s
- Bottleneck router buffer size = **18 packets**
 - Throughput = 9.99 Mb/s
 - Goodput = 9.99 Mb/s
 - Packet loss rate = 0.00192
 - RTT = 0.030 s
- Bottleneck router buffer size = **9 packets**
 - Throughput = 9.45 Mb/s
 - Goodput = 9.42 Mb/s
 - Packet loss rate = $26 / (9355 - 1482) = 0.00330$
 - RTT = 0.024 s

Sizing Router Buffers (cont'd)

- Edit: **tcp.tcl**
- Bottleneck router buffer size = **2 packets**



```
*tcp.tcl
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

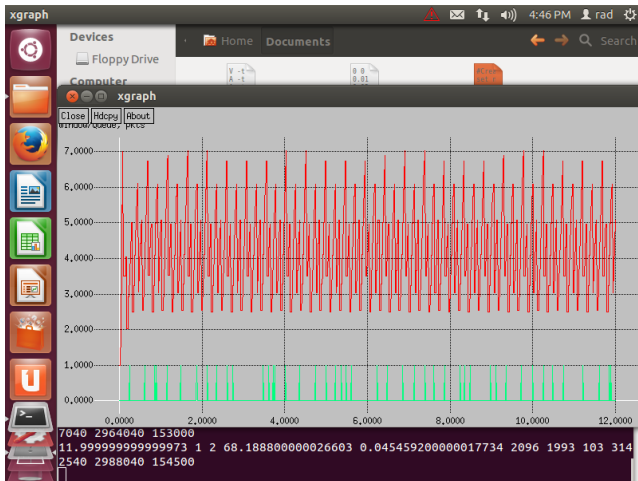
#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 2
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
$tcp attach $trace
```

Sizing Router Buffers (cont'd)

- Time: **from 2 to 12**
- Throughput = ? Packet loss rate = ? Goodput = ? Mean RTT = ?

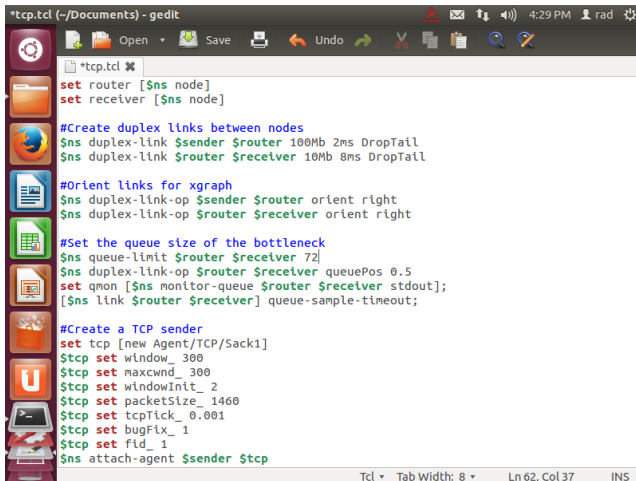


Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = **18 packets**
 - Throughput = 9.99 Mb/s
 - Goodput = 9.99 Mb/s
 - Packet loss rate = 0.00192
 - RTT = 0.030 s
- Bottleneck router buffer size = **9 packets**
 - Throughput = 9.45 Mb/s
 - Goodput = 9.42 Mb/s
 - Packet loss rate = 0.00330
 - RTT = 0.024 s
- Bottleneck router buffer size = **2 packets**
 - Throughput = 2.11 Mb/s
 - Goodput = 2.01 Mb/s
 - Packet loss rate = $83 / (2098 - 339) = 0.04719$
 - RTT = 0.021 s

Sizing Router Buffers (cont'd)

- Edit: **tcp.tcl**
- Bottleneck router buffer size = **72 packets**



```
*tcp.tcl ✖
set router [$ns node]
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

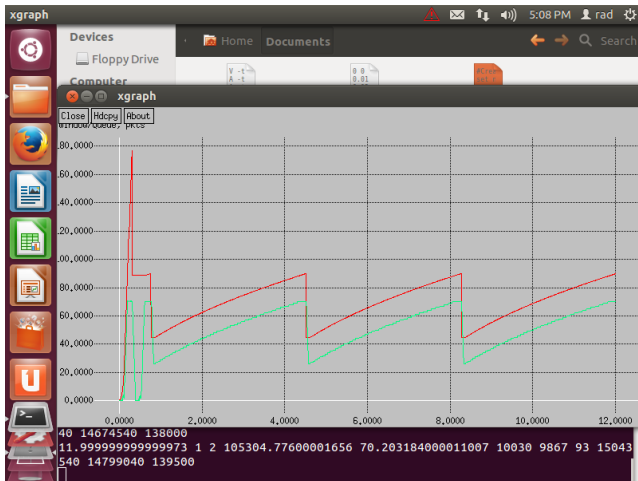
#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 72
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
```

Tcl ▾ Tab Width: 8 ▾ Ln 62, Col 37 INS

Sizing Router Buffers (cont'd)

- Time: **from 2 to 12**
- Throughput = ? Packet loss rate = ? Goodput = ? Mean RTT = ?

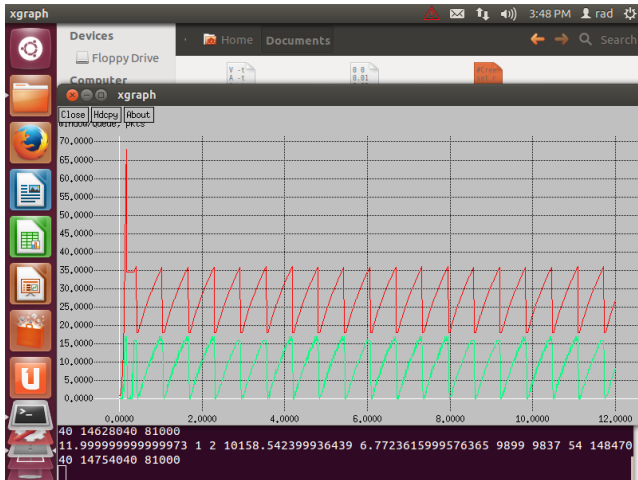


Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = **72 packets**
 - Throughput = 10.03 Mb/s
 - Goodput = 10.03 Mb/s
 - Packet loss rate = $2 / (10032 - 1670) = 0.00024$
 - RTT = 0.081 s
- Bottleneck router buffer size = **36 packets**
 - Throughput = 10.02 Mb/s
 - Goodput = 10.02 Mb/s
 - Packet loss rate = 0.00084
 - RTT = 0.046 s
- Bottleneck router buffer size = **18 packets**
 - Throughput = 9.99 Mb/s
 - Goodput = 9.99 Mb/s
 - Packet loss rate = 0.00192
 - RTT = 0.030 s

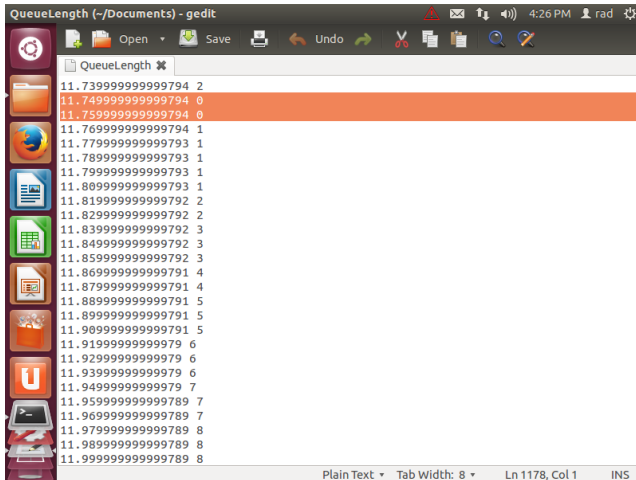
Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = 18 packets



Sizing Router Buffers (cont'd)

- If the buffer never goes empty, the router must be sending packets onto the bottleneck link at the maximum rate of this link



QueueLength (~/.Documents) - gedit

Open Save Undo

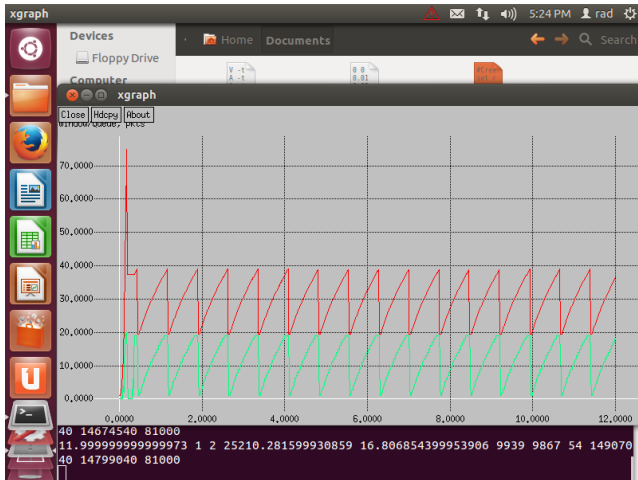
QueueLength

```
11.739999999999794 2
11.749999999999794 0
11.759999999999794 0
11.769999999999794 1
11.779999999999793 1
11.789999999999793 1
11.799999999999793 1
11.809999999999793 1
11.819999999999792 2
11.829999999999792 2
11.839999999999792 3
11.849999999999792 3
11.859999999999792 3
11.869999999999791 4
11.879999999999791 4
11.889999999999791 5
11.899999999999791 5
11.909999999999791 5
11.91999999999979 6
11.92999999999979 6
11.93999999999979 6
11.94999999999979 7
11.959999999999789 7
11.969999999999789 7
11.979999999999789 8
11.989999999999789 8
11.999999999999789 8
```

Plain Text Tab Width: 8 Ln 1178, Col 1 INS

Sizing Router Buffers (cont'd)

- Bottleneck router buffer size = **21 packets**



Sizing Router Buffers (cont'd)

- Overbuffering is a bad idea!



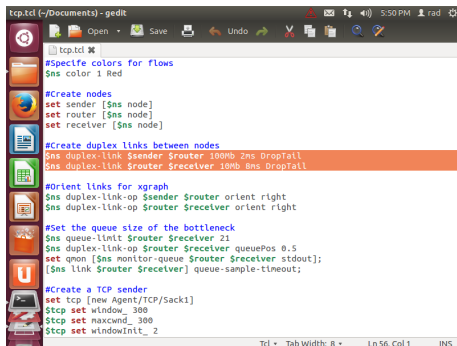
Outline

- 1 VMware Player
- 2 Sizing router buffers
- 3 Bandwidth-delay product

Bandwidth-Delay Product

- **Bandwidth-Delay Product (BDP)** =
= bottleneck link speed (in b/s) $\times$ RTT (in seconds)

$$\text{BDP} = \frac{10 \cdot 10^6 \times 20 \cdot 10^{-3}}{1500 \times 8} \approx 18$$



```
tcp.tcl (-/Documents) - gedit
tcp.tcl
#Specify colors for flows
$ns color 1 Red

#Create nodes
set sender [$ns node]
set router [$ns node]
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

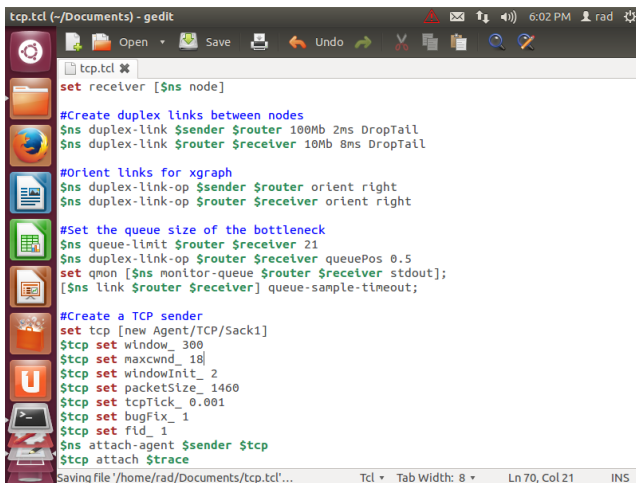
#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 21
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
```

Sizing Router Buffers (cont'd)

- $rwnd = BDP$



The screenshot shows a Gedit window titled 'tcp.tcl (~/Documents) - gedit'. The window contains a Tcl script for setting up a network simulation. The script includes comments and commands for creating duplex links, orienting links, setting queue sizes, and creating a TCP sender. The status bar at the bottom indicates 'Saving file '/home/rad/Documents/tcp.tcl...', 'Tcl', 'Tab Width: 8', 'Ln 70, Col 21', and 'INS'.

```
tcp.tcl (~/Documents) - gedit
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

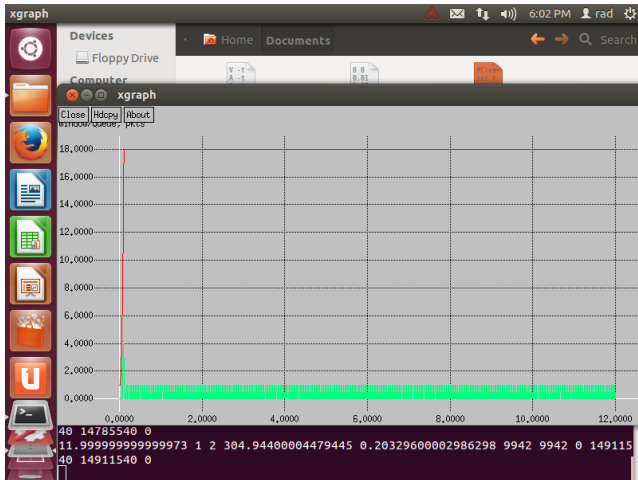
#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 21
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 18
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
$tcp attach $trace
```

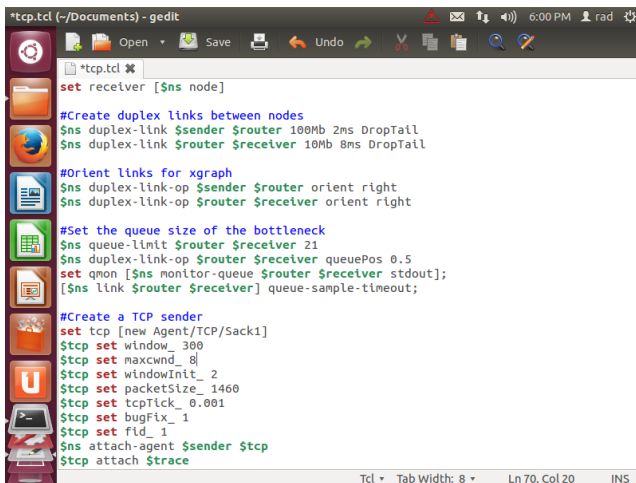
Sizing Router Buffers (cont'd)

- Nor buffer overflows, nor large queues



Sizing Router Buffers (cont'd)

- $rwnd < BDP$



The screenshot shows a Gedit editor window titled '*tcp.tcl (~/.Documents) - gedit'. The window contains a Tcl script for setting up a network simulation. The script includes comments and commands for creating duplex links, orienting links, setting queue limits, and creating a TCP sender. The status bar at the bottom indicates 'Tcl', 'Tab Width: 8', 'Ln 70, Col 20', and 'INS'.

```
*tcp.tcl (~/.Documents) - gedit
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

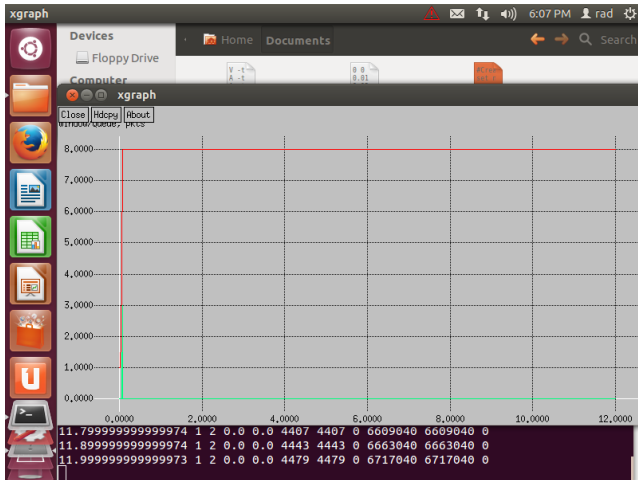
#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 21
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 300
$tcp set maxcwnd_ 8
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
$tcp attach $trace
```

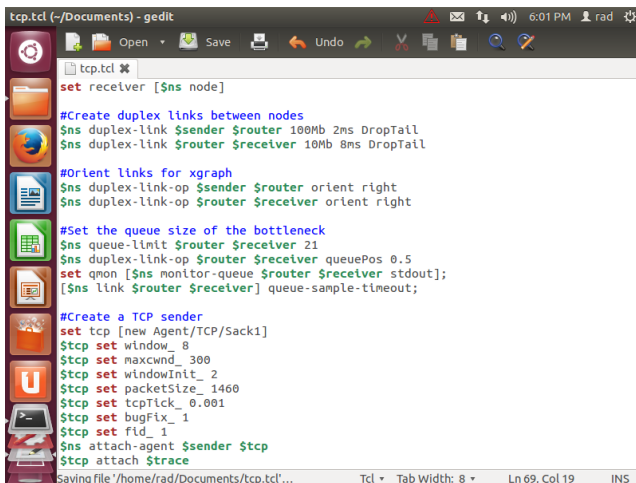
Sizing Router Buffers (cont'd)

- Under-utilization of the bottleneck link capacity



Sizing Router Buffers (cont'd)

- $ssthresh \ll BDP$



```
tcp.tcl (~/Documents) - gedit
set receiver [$ns node]

#Create duplex links between nodes
$ns duplex-link $sender $router 100Mb 2ms DropTail
$ns duplex-link $router $receiver 10Mb 8ms DropTail

#Orient links for xgraph
$ns duplex-link-op $sender $router orient right
$ns duplex-link-op $router $receiver orient right

#Set the queue size of the bottleneck
$ns queue-limit $router $receiver 21
$ns duplex-link-op $router $receiver queuePos 0.5
set qmon [$ns monitor-queue $router $receiver stdout];
[$ns link $router $receiver] queue-sample-timeout;

#Create a TCP sender
set tcp [new Agent/TCP/Sack1]
$tcp set window_ 8
$tcp set maxcwnd_ 300
$tcp set windowInit_ 2
$tcp set packetSize_ 1460
$tcp set tcpTick_ 0.001
$tcp set bugFix_ 1
$tcp set fid_ 1
$ns attach-agent $sender $tcp
$tcp attach $trace

Saving File '/home/rad/Documents/tcp.tcl'...
Tcl Tab Width: 8 Ln 69, Col 19 INS
```

Sizing Router Buffers (cont'd)

- Prematurely switching from slow start to congestion avoidance

